Title
Improved Small-Sample Estimation of Nonlinear Cross-Validated Prediction Metrics

Journal
Journal of the American Statistical Association, 115(532)

ISSN
0162-1459

Authors
Benkeser, David
Petersen, Maya
van der Laan, Mark J

Publication Date
2020-10-01

DOI
10.1080/01621459.2019.1668794

Peer reviewed

HHS Public Access

Author manuscript

J Am Stat Assoc. Author manuscript; available in PMC 2021 March 12.

Published in final edited form as:

J Am Stat Assoc. 2020 ; 115(532): 1917–1932. doi:10.1080/01621459.2019.1668794.

Improved small-sample estimation of nonlinear cross-validated prediction metrics

David Benkeser[1], **Maya Petersen**[2], **Mark J van der Laan**[2,3]

[1]Department of Biostatistics and Bioinformatics, Emory University

[2]Graduate Group in Biostatistics, University of California, Berkeley

[3]Department of Statistics, University of California, Berkeley

Abstract

When predicting an outcome is the scientific goal, one must decide on a metric by which to evaluate the quality of predictions. We consider the problem of measuring the performance of a prediction algorithm with the same data that were used to train the algorithm. Typical approaches involve bootstrapping or cross-validation. However, we demonstrate that bootstrap-based approaches often fail and standard cross-validation estimators may perform poorly. We provide a general study of cross-validation-based estimators that highlights the source of this poor performance, and propose an alternative framework for estimation using techniques from the efficiency theory literature. We provide a theorem establishing the weak convergence of our estimators. The general theorem is applied in detail to two specific examples and we discuss possible extensions to other parameters of interest. For the two explicit examples that we consider, our estimators demonstrate remarkable finite-sample improvements over standard approaches.

Keywords

prediction; machine learning; cross-validation; AUC; targeted minimum loss-based estimation; estimating equations

1 Introduction

Prediction is important in many areas of research. For example, in medical applications, algorithms can predict prognoses for patients, allowing clinicians to better weigh risks and benefits of different treatments. Modern technology allows for collection of vast amounts of data, including genetic sequences, gene expressions, proteomics, and metabolomics. Relative to the amount of information measured on each patient, the total number of patients available may be quite modest. Many practical applications thus require methodology that enables researchers to simultaneously develop and evaluate prediction algorithms in small samples.

Corresponding author David Benkeser benkeser@emory.edu.

Myriad tools are available for developing prediction algorithms. These tools range from classical approaches like logistic regression to algorithms developed in the machine learning literature such as deep neural networks. In addition to choosing a method for creating a prediction algorithm, researchers must select a metric that quantifies performance. The metric should be informed by the scientific context and many metrics have been proposed (Steyerberg et al., 2010). The performance metric of a given algorithm predicting on new data is also referred to as its test error, generalization error, or conditional error (Friedman et al., 2001) and can be viewed as a summary of a population of interest. For example, a commonly used metric for predicting the onset of a clinical disease is the area under the receiver operating characteristic curve (AUC). AUC describes the probability that a randomly selected individual from the population who will develop disease (hereafter, case) has a higher predicted risk of disease than a randomly selected individual who *will not* develop disease (*control*). It is implicit in this definition that cases and controls are sampled from some population. The AUC of the same prediction algorithm may be quite different when considering drawing individuals from different populations. In this work, we consider prediction metrics defined with respect to the population from which the sample data were generated. For discussion on generalization, see Glümer et al. (2006); Moons et al. (2012), and Austin et al. (2017).

It is well known that evaluating the performance of an algorithm using the same data used to train the algorithm can lead to an optimistic estimate of performance. To correct for this bias, it is necessary to employ one of several optimism-correcting techniques. Common approaches include bootstrapping, single sample splitting, and K-fold cross-validation (CV). Here, we provide a short description of how each approach arrives at a final prediction algorithm and an assessment of its performance. Several bootstrap corrections have been described. For example, Harrell et al. (1996) recommends training a prediction algorithm using all observations. To obtain an estimate of the performance, one repeatedly samples with replacement from the observations and trains the algorithm on each re-sampled data set. The performance of the algorithm trained on the re-sampled data is estimated using both the re-sampled data and the original data. The average difference between the two over many re-samples estimates the bias of the prediction metric. The performance of the algorithm trained using the entire data set is estimated using those same data, and the final estimate of predictive performance is the difference between this metric estimated using all of the data and the bootstrap-estimated bias. Friedman et al. (2001) describes alternative bootstrap procedures. For single sample splitting, data are randomly partitioned into two groups. The prediction algorithm is trained using one of the groups (the training set), and this is the final algorithm that is reported. The prediction performance of the algorithm trained in the training sample is estimated in the held-out group (the test set). K-fold CV generalizes single sample splitting by partitioning the data into several distinct groups. The prediction algorithm is developed using all but one group, and the prediction metric is estimated in the remaining group. The process is repeated until each group has been used to estimate the prediction metric once. The optimism-corrected estimate of the metric is the average of these estimates. It is common to then train the algorithm using the entire data set and report this algorithm as the final prediction algorithm.

Each approach has relative strengths and weaknesses. A strength of single sample splitting is that the final prediction algorithm is evaluated directly. However, this same strength may be viewed as a weakness, in that each observation is used either to train the algorithm or to estimate its performance. Thus, in small samples the procedure is inefficient both in generating the best-performing prediction algorithm and in estimating its performance because it only uses part of the data for each task. In contrast, bootstrap approaches allow the use of the entire data set to train the prediction algorithm and to estimate the prediction metric. While these approaches have been validated empirically for smooth prediction functions such as logistic regression (Steyerberg et al., 2001; Smith et al., 2014), they often lack a theoretical basis in the context of modern machine learning algorithms.

K-fold CV also uses all observations to train and evaluate a prediction algorithm; however, in contrast to bootstrapping, theoretical frameworks have been developed that apply when considering arbitrary learning algorithms (Hubbard et al., 2016). Moreover, it is often possible to construct closed-form, computationally efficient confidence intervals and hypothesis tests based on K-fold CV estimates (e.g., LeDell et al. (2015)). Nevertheless, some researchers are skeptical of this approach, since it relies on the assumption that the average performance of the algorithm trained using $(K-1)/K \times 100\,\%$ of the data accurately reflects the performance of the algorithm trained using the entire data set. This fact draws attention to the important issue of selecting K. Including a larger percentage of the data in the training sample leads to greater similarity between the training-sample-specific algorithms and the algorithm trained using all observations. On the other hand, using more data to train the algorithm leaves less data for estimating the prediction metric. Thus, we might expect greater variance in the estimates of performance.

While there has been some work on how to choose K for prediction metrics that are linear in the data generating distribution, such as mean squared-error and mean absolute-deviation (e.g., Chapter 7 of Friedman et al. (2001)), there are limited references available for nonlinear prediction metrics, such as quantiles of absolute deviation, AUC, and time-varying AUC (Heagerty et al., 2000). Whereas linear metrics can be estimated using estimators that themselves are linear, nonlinear metrics generally require estimators that are *asymptotically linear*; that is, they behave as an empirical average plus a remainder term. While the remainder is often negligible in large samples, in finite samples it may contribute substantially to the behavior of the estimator. Thus, when nonlinear prediction metrics are of interest, we must carefully consider selection of K and estimation strategies for the prediction metric.

In this work, we propose three general estimation strategies for nonlinear K-fold CV prediction metrics. These strategies are tailored for finite-sample performance, while retaining desirable asymptotic properties. The estimators' construction is informed by viewing the performance metric as a statistical functional of several nuisance parameters. For example, in the case of AUC, the relevant nuisance parameters are the cumulative distribution function of predictions made amongst cases and amongst controls. Standard K-fold CV estimators use estimates of these quantities based on the (potentially small) validation sample, while our proposed estimators use estimates based on the (potentially much larger) training sample. A second stage bias correction is applied to account for

potential over-fitting. While each of the three estimators we propose follows this general approach, they differ in their means of bias correction: we consider correcting bias using (i) estimating equations; (ii) a one-step Newton-Raphson approach; and (iii) targeted minimum loss-based estimation. We provide a general theorem establishing the weak convergence of these estimators that applies to a large class of smooth prediction metrics, and discuss how the theorem can be used to build asymptotically justified confidence intervals and hypothesis tests.

As illustration, we apply our general theorem to a number of specific metrics. The first is AUC, described above. The second is a parameter called the sensitivity-constrained rate of negative prediction (SCRNP). This quantity describes the probability of classifying an observation as a control under the constraint that a (user-specified) high percentage of cases are correctly classified. To understand the relevance of this metric, consider developing an algorithm for predicting breast cancer incidence in women. We should like to ensure that we identify a large proportion of women who will eventually develop breast cancer; that is, we would like to enforce that our procedure for selecting high-risk women has high sensitivity. However, women with high predicted risk of cancer may be recommended to undergo more invasive screening procedures. So beyond our sensitivity constraint, we should like to maximize the proportion of women who are not required to undergo additional screening. The SCRNP describes this proportion. Zheng et al. (2018) discuss SCRNP in the context of HIV prevention. Application of our general theorem leads to novel efficiency theory for the SCRNP parameter, and the resultant estimators show remarkable finite-sample improvements relative to existing approaches in both simulated and real data analysis. Because this metric offers a practically relevant way of evaluating prediction algorithms, we suggest the theory of its estimation and inference is an important contribution in its own right. In addition to these two detailed examples, we discuss application of our theorem to other prediction metrics. In sum, our main contribution is three new strategies for estimating a large class of CV-based prediction metrics together with a general theory establishing the weak convergence properties of these estimators, thereby proving a basis for valid statistical inference. We demonstrate application of this framework to a number of metrics, including AUC and SCRNP, and discuss its use to select tuning parameters of prediction algorithms. Finally, we provide open source software that implements all the estimators that we describe.

2 Preliminaries

2.1 Notation and parameter of interest

Suppose the data consist of n independent realizations of a random variable $O = (X, Y) \sim P_0$, where $X \in \mathcal{X}$ is a vector of predictors, $Y \in \mathcal{Y}$ is an outcome, and P_0 is the unknown true distribution of O. We denote by P_n the empirical measure of $(O_1, \ldots, O_n)$, by $\mathcal{O}$ the support of O, and by $\mathcal{M}$ a nonparametric model for P_0. We focus on situations in which the outcome is binary $\mathcal{Y} = \{0, 1\}$, though our theory applies equally well to arbitrary $\mathcal{Y}$. We use *case* to refer to an observation with $Y = 1$ and *control* to refer to an observation with $Y = 0$. We use $\widehat{\Psi} : \mathcal{M} \to \psi$ to denote a method for training a prediction algorithm. Because our focus is on binary prediction problems, we assume, without loss of generality, that ψ is the

space of functions mapping from $\mathcal{X}$ to [0,1]. We denote by $\psi_n = \widehat{\Psi}(P_n)$ the prediction algorithm trained using $(O_1, \ldots, O_n)$. As stated previously, we consider a procedure for creating and evaluating a prediction algorithm wherein the algorithm is trained using all of the observations. Thus, our procedure ultimately returns ψ_n as the prediction algorithm, along with a measure of its predictive performance.

For a given $\psi : \mathcal{X} \to \psi$, we denote by $\Phi_\psi : \mathcal{M} \to \Phi$ a prediction metric of interest. For each $P \in \mathcal{M}$, write to $\Phi_\psi(P)$ denote the value of this metric implied by P; that is, $\Phi_\psi(P)$ describes how well ψ predicts when sampling from a population with a particular distribution P of O. This notation makes explicit the point made in the introduction regarding a prediction function's performance being specific to the population. The parameter Φ_ψ quantifies the performance of ψ as a function of P, the population distribution of O.

It is useful for us to regard Φ_ψ as a functional of a vector of nuisance parameters $Q_\psi(P) := \{Q_{\psi,1}(P), \ldots, Q_{\psi,M}(P)\}$, rather than P itself. Therefore, to simplify notation, we will often write $\Phi_\psi(Q_\psi)$ as shorthand for $\Phi_\psi(Q_\psi(P))$ for a general $P \in \mathcal{M}$. we introduce the shorthand $Q_{0,\psi} := Q_\psi(P_0)$ denote the value of the nuisance parameter under P_0.

We wish to study performance of the algorithm ψ_n under P_0, the distribution of the population from which our observations were sampled. Thus, our goal is to estimate $\phi_{0,\psi_n} := \Phi_{\psi_n}(Q_{0,\psi_n})$. We develop general theory for estimation of any prediction metric that satisfies a particular smoothness criterion; the metric must be *pathwise differentiable* for each $P \in \mathcal{M}$. That is, when considering a smooth one-dimensional parametric submodel $\{P_\epsilon : \epsilon\} \subset \mathcal{M}$ through P with score $\mathcal{S}$ at $\epsilon = 0$ amongst a rich class of such submodels, we have that $\frac{d}{d\epsilon}\Phi_\psi(P_\epsilon)|_{\epsilon=0}$ exists and can be represented as $\frac{d}{d\epsilon}\Phi_\psi(P_\epsilon)|_{\epsilon=0} = \mathrm{E}_P\{D_\psi(P)(O)S(O)\}$, where $D_\psi(P)$ is the unique gradient of Φ_ψ at P in a nonparametric model (see e.g., Chapter 25 of van der Vaart (2000)). Beyond this explicit condition, pathwise differentiability implies that Φ_ψ depends on $dP(o)$ through $o \in \mathcal{S}$ for some non-zero measure set $\mathcal{S}$. Thus, our theorem *will not* generally apply to estimation of parameters such as the prediction error of ψ at a given point x, e.g., as measured by the conditional mean of $|Y - \psi(X)|$ given $X = x$.

2.2 Cross-validation

Consider an estimator $\widehat{Q}_\psi : \mathcal{M} \to \mathbb{Q}$ of $Q_{0,\psi} := Q_\psi(P_0)$, where $\mathbb{Q} = \mathbb{Q}_1 \times \ldots \times \mathbb{Q}_M$ is the parameter space of $Q_{0,\psi}$. Define $Q_{n,\psi} := \widehat{Q}_\psi(P_n)$ as an estimate of $Q_{0,\psi}$ based on $(O_1, \ldots, O_n)$. A naïve estimate of ϕ_{0,ψ_n} is $\Phi_{\psi_n}(Q_{n,\psi_n})$, which uses all of the observations to train $\widehat{\Psi}$ and to evaluate its performance. It is well known that this naïve estimate is almost always biased for ϕ_{0,ψ_n}. This motivates the use of CV. We consider K-fold CV, a process that consists of partitioning the data into K blocks. A single block, the so-called *validation fold*, is withheld and the prediction algorithm is developed on the data in the remaining $K-1$ blocks, the so-called *training fold*. The prediction metric for the prediction algorithm developed in the training fold is estimated using the validation fold. The process is repeated

K times until each block has been validation fold once, and the K estimates are averaged to obtain a final estimate. We notate this process as follows. For $k = 1, \ldots, K$, we define $P^1_{n,k}$ and $P^0_{n,k}$ to be the empirical distributions of the k-th validation and training fold, respectively. We denote by $\psi^0_{n,k} := \widehat{\Psi}\left(P^0_{n,k}\right)$ the prediction algorithm developed in the k-th training fold. The CV version of the prediction metric parameter is $\Phi_{\text{cv}}(P) := \frac{1}{K}\sum_{k=1}^{K} \Phi_{\psi^0_{n,k}}(P)$. Whether or not $\phi_{0,\text{cv}} := \Phi_{\text{cv}}(P_0)$ provides a good approximation of ϕ_{0,ψ_n} depends on how well the performance of $\psi^0_{n,k}$ approximates the performance of ψ_n. If for each k, the performance of the prediction algorithm trained using training data is similar to that of the prediction algorithm trained using the entire data set, then we might expect near equality of ϕ_{0,ψ_n} and $\phi_{0,\text{cv}}$. We would also expect that estimators of $\phi_{0,\text{cv}}$ will perform well for the sake of estimating ϕ_{0,ψ_n}. Thus, our strategy for estimating $\phi_{0.\psi_n}$ is to obtain instead an estimator of $\phi_{0,\text{cv}}$.

2.3 Standard CV-based estimators

The standard approach to estimation of $\phi_{0,\text{cv}}$ uses the k-th training sample to obtain $\psi^0_{n,k}$ and the k-th validation sample to estimate $\Phi_\psi(P_0)$, e.g., the plug-in estimator

$$\phi_{n,\text{cv}} := \frac{1}{K}\sum_{k=1}^{K} \Phi_{\psi^0_{n,k}}\left(Q^1_{n,k}\right), \tag{1}$$

where $Q^1_{n,k} := \widehat{Q}_{\psi^0_{n,k}}\left(P^1_{n,k}\right)$ is an estimate of $Q_{0,k} := Q_{0,\psi^0_{n,k}}$ based on the k-th validation sample. An explicit example of a standard K-fold CV estimator of AUC is provided in the web supplement. Estimators of the form (1) are ubiquitous in the literature, but face a "bias-variance" trade-off when considering estimation of ϕ_{0,ψ_n}. These estimators may be biased, in that the true value of the theoretical CV parameter $\phi_{0,\text{cv}}$ may differ largely from ϕ_{0,ψ_n}. This "bias" may be mitigated by selecting large K, which generates larger training folds and more stability in the prediction algorithms across folds. However, the small validation samples may impart increased variability to the estimates $Q^1_{n,k}, k = 1, \ldots, K$, and thus more variability to $\phi_{n,\text{cv}}$. Therefore, depending on how K is selected, $\phi_{n,\text{cv}}$ may perform poorly either in terms of bias or variance. As we show below, whether or not this is the case depends on the analysis of the prediction metric as a statistical functional.

Remark on single sample splitting: The single sample split approach splits the data into a single training and single test set. A common recommendation is to use 70% of the data for training sample and 30% for validation sample. We use $P^0_{n,s}$ and $P^1_{n,s}$ to denote the empirical distribution of the training and validation samples, respectively. We denote by $\psi^0_{n,\text{s}} := \widehat{\Psi}\left(P^0_{n,\text{s}}\right)$ the prediction algorithm developed in the training fold. The sample split target parameter for a given $P \in \mathcal{M}$ is $\Phi_{\psi^0_{n,s}}(P)$. A natural plug-in estimator of this quantity is $\Phi_{\psi^0_{n,k}}\left(Q^1_{n,\text{s}}\right)$, where $Q^1_{n,\text{s}} := \widehat{Q}_{\psi^0_{n,\text{s}}}\left(P^1_{n,\text{s}}\right)$ is an estimate of $Q_{0,\text{s}} := Q_{0,\psi^0_{n,s}}$ based on the

validation sample. An explicit example of a sample-splitting estimator of AUC is provided in the web supplement.

Analysis of standard CV-based estimators

Pathwise differentiability of Φ_ψ implies the following linear expansion of Φ_ψ at $P \in \mathcal{M}$,

$$\Phi_\psi\left(Q_\psi\right) - \Phi_\psi\left(Q_{0,\psi}\right) = -P_0 D_\psi\left(Q_\psi, G_\psi\right) + R_\psi\left(Q_\psi, Q_{0,\psi}, G_\psi, G_{0,\psi}\right), \tag{2}$$

where $D_\psi\left(Q_\psi, G_\psi\right)$ is the unique gradient of Φ_ψ at P in a nonparametric model (Pfanzagl, 1982). The second term, $R_\psi\left(Q_\psi, Q_{0,\psi}, G_\psi, G_{0,\psi}\right)$, is a second-order remainder, which will presently become our focus. In (2), we introduced the notation $G_\psi := G_\psi(P) \in \mathcal{G}$ to denote nuisance parameters in addition to Q_ψ that may appear in the gradient. For example, the gradient of the AUC parameter involves $G_\psi = g_P$. Because Φ_ψ only depends on P through Q_ψ, (2) holds for any choice of G_ψ that is implied by some distribution in $\mathcal{M}$. In other words, we may allow $G_\psi = G_\psi(P')$ for $P' \neq P$. Suppose that an estimator $\widehat{G}_\psi : \mathcal{M} \to \mathcal{G}$ is available for any given ψ. We define $G^1_{n,k} := \widehat{G}^0_{\psi_{n,k}}\left(P^1_{n,k}\right)$ and $G_{0,k} := G^0_{\psi_{n,k}}(P_0)$. We apply (2) to the estimator (1) for each CV fold,

$$\begin{aligned}
\phi_{n,\mathrm{cv}} - \phi_{0,\mathrm{cv}} &= \frac{1}{K}\sum_{k=1}^{K}\left\{\Phi^0_{\psi_{n,k}}\left(Q^1_{n,k}\right) - \Phi^0_{\psi_{n,k}}\left(Q_{0,k}\right)\right\} \\
&= \frac{1}{K}\sum_{k=1}^{K}\left\{-P_0 D^0_{\psi_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}\right) + R^0_{\psi_{n,k}}\left(Q^1_{n,k}, Q_{0,k}, G^1_{n,k}, G_{0,k}\right)\right\} \\
&= \frac{1}{K}\sum_{k=1}^{K}\left[\left(P^1_{n,k} - P_0\right) D^0_{\psi_{n,k}}\left(Q_{0,k}, G_{0,k}\right) - P^1_{n,k} D^0_{\psi_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}\right)\right. \\
&\quad + \left(P^1_{n,k} - P_0\right)\left\{D^0_{\psi_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}\right) - D^0_{\psi_{n,k}}\left(Q_{0,k}, G_{0,k}\right)\right\} \\
&\quad \left. + R^0_{\psi_{n,k}}\left(Q^1_{n,k}, Q_{0,k}, G^1_{n,k}, G_{0,k}\right)\right].
\end{aligned} \tag{3}$$

We examine each term of (3) in turn. The first term is the empirical average of the centered gradient in the validation sample; that is, an empirical average of independent mean-zero terms with finite variance. Thus, standard statistical tools such as the weak law of large numbers and central limit theorem may be applied to study the stochastic behavior of this term. The second term is the empirical average in the validation sample of the gradient at the values of the nuisance parameters estimated in the validation sample. If $Q^1_{n,k}$ is a maximum likelihood estimator, then this term will naturally equal zero. This is true, for example, of AUC, where the relevant nuisance parameters are cumulative distribution functions, which may be estimated via their empirical counterparts. However, more generally, prediction metrics may require nonparametric smoothing in estimation of $Q_{0,k}$ or $G_{0,k}$. This occurs, for example, when the metric of interest pertains to an individualized treatment rule (Luedtke and van der Laan, 2016), or to time-varying AUC when outcome measurements may be subject to covariate-dependent right censoring. In these cases, because the validation data are used both to estimate the relevant nuisance parameters, as well as to evaluate the

gradient, this term will generally have irregular behavior, resulting in large bias of the CV estimator. Several techniques are available in the literature to correct for such behavior. These include one-step (Ibragimov and Has'minskii, 1981; Pfanzagl, 1982), estimating equations (van der Laan and Robins, 2003), and targeted minimum loss-based estimation (TMLE) corrections (van der Laan and Rubin, 2006). These techniques are discussed in detail in Appendix A. The third term is referred to as an empirical process term as it involves the difference between $P^1_{n,k}$ and P_0 applied to the difference between gradient at the estimated and true nuisance parameters. This term can be shown to be $o_p\left(n^{-1/2}\right)$ if

$D^0_{\psi^0_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}\right)$ falls in a P_0 -Donsker class with probability tending to one and

$$P_0\{D^0_{\psi^0_{n,k}}(Q^1_{n,k}, G^1_{n,k}) - D^0_{\psi^0_{n,k}}(Q_{0,k}, G_{0,k})\}^2 \to 0 \text{ in probability as } n \to \infty . \quad (4)$$

The final term is the second-order remainder, which often involves P_0 applied to differences between estimated nuisance parameters and true nuisance parameters. For example, many remainders exhibit *doubly-robust* structure and involve P_0 applied to a product of $Q^1_{n,k} - Q_{0,k}$ and $G^1_{n,k} - G_{0,k}$. Because of this cross-product structure, under mild conditions the remainder converges to zero in probability if either $Q^1_{n,k}$ or $G^1_{n,k}$ is consistent for $Q_{0,k}$ and $G_{0,k}$, respectively. Moreover, we are often able to study the form of the remainder to provide sufficient conditions on nuisance estimators to ensure that $R^0_{\psi^0_{n,k}}(Q^1_{n,k}, Q_{0,k}, G^1_{n,k}, G_{0,k}) = o_p(n^{-1/2})$.

While the remainder and empirical process term are often *asymptotically* negligible, in finite samples they can contribute substantially to the behavior of $\phi_{n,\text{cv}}$. In K-fold CV, the nuisance estimates $Q^1_{n,k}$ and $G^1_{n,k}$ are based only on approximately n/K observations. Thus, if K is large, n is small, or both, the nuisance estimates may not provide a good approximation of their true values, which in turn has a deleterious effect on the finite-sample behavior of $\phi_{n,\text{cv}}$. We conclude by noting that the expansion (2) naturally facilitates study of the single sample splitting estimator as well, and that this same discussion applies to that estimator.

Remark:

A possible exception to the above discussion is metrics that are linear in P. For example, given ψ, consider $\Phi_\psi(P) := \mathrm{E}_P\{\mathbb{L}_\psi(O)\}$, where $(\psi, O) \mapsto \mathbb{L}_\psi(O)$ is a some measure of the distance between $\psi(X)$ and Y, such as squared error loss $\mathbb{L}_\psi(O) = \{Y - \psi(X)\}^2$. The gradient of such a parameter with respect to $\mathcal{M}$ is $D_\psi(P)(O_i) = \mathbb{L}_\psi(O_i) - \mathrm{E}_P\{\mathbb{L}_\psi(O_i)\}$. Given ψ and a sample of data with empirical distribution P^*_n, we may estimate $\Phi_\psi(P_0)$ by $\mathrm{E}_{p^*_n}\{\mathbb{L}_\psi(O)\}$. Thus, $\Phi_\psi(P^*_n) - \Phi_\psi(P_0)$ writes as

$$\mathrm{E}_{P^*_n}\{\mathbb{L}_\psi(O)\} - \mathrm{E}_{P_0}\{\mathbb{L}_\psi(O)\} = -\mathrm{E}_{P_0}[\mathbb{L}_\psi(O) - \mathrm{E}_{P^*_n}\{\mathbb{L}_\psi(O)\}] = -P_0 D_\psi(P^*_n),$$

and (2) holds with the second-order remainder equal to zero. By extension, the CV estimator $\phi_{n,\mathrm{cv}}$ behaves as an empirical average without the need to introduce empirical process terms. Therefore, our analysis suggests that for estimators of linear prediction metrics, choosing large K may prove to be the best strategy. Indeed, the optimal strategy may involve letting K grow quickly with n, as in leave-one-out CV. However, analysis of these estimators is challenging as the above discussion relies on the size of the validation sample growing to infinity. While some progress on this problem has been made with respect to convergence rates of certain leave-one-out estimators (Kandasamy et al., 2015; Benkeser et al., 2017), to our knowledge, no general theory of weak convergence has been presented in the literature.

4 Alternative CV-based estimators

Rather than basing our estimates of $Q_{0,k}$ and $G_{0,k}$ on the validation sample, we propose to generate these estimates based on the training sample, and to subsequently use the validation sample to control the bias of the estimates using extensions of the techniques described in Appendix A. We define $Q^0_{n,k} := \widehat{Q}_{\Psi^0_{n,k}}\left(P^0_{n,k}\right)$ and $G^0_{n,k} := \widehat{G}\left(P^0_{n,k}\right)$ to be training sample estimates of $Q_{0,k}$ and $G_{0,k}$, respectively. Consider the estimator

$$\phi^{\dagger}_{n,\mathrm{cv}} := \frac{1}{K}\sum_{k=1}^{K} \Phi_{\Psi^0_{n,k}}\left(Q^0_{n,k}\right). \tag{5}$$

of $\phi_{0,\mathrm{cv}}$. We argued in the introduction that such an estimator would be biased. Nevertheless, we can apply (2) to this estimator, which allows us to express $\phi^{\dagger}_{n,\mathrm{cv}} - \phi_{0,\mathrm{cv}}$ as

$$\begin{aligned}
&\frac{1}{K}\sum_{k=1}^{K}\left[(P^1_{n,k}-P_0)D_{\Psi^0_{n,k}}(Q_{0,k},G_{0,k}) - P^1_{n,k}\{D_{\Psi^0_{n,k}}(Q^0_{n,k},G^0_{n,k})\}\right.\\
&+(P^1_{n,k}-P_0)\{D_{\Psi^0_{n,k}}(Q^0_{n,k},G^0_{n,k}) - D_{\Psi^0_{n,k}}(Q_{0,k},G_{0,k})\}\\
&\left.+R_{\Psi^0_{n,k}}(Q^0_{n,k},Q_{0,k},G^0_{n,k},G_{0,k})\right].
\end{aligned} \tag{6}$$

Comparing equation (3) to (6), we find the first term to be unchanged. The other terms in (6) are identical to (3), but the nuisance parameters are now estimated using the training sample, which affords us two putative benefits. The first is that assumption (4) is no longer necessary, since, conditioning on the k-th training sample, the third term in (6) is an empirical average of independent and identically distributed random variables. The second benefit is that the remainder term, which often involves products of differences between estimated and true nuisance parameters, includes estimated nuisance parameters that are based on $n(K-1)/K$, rather than n/K, observations. Therefore, if we can use the training samples to generate reasonable estimates of $Q_{0,k}$ and $G_{0,k}$, then can we expect the remainder will exhibit improved behavior when K is large, n is small, or both. However, a potential challenge to the performance of $\phi^{\dagger}_{n,\mathrm{cv}}$ is that the second term in the sum must be accounted for in order to prove weak convergence. We can apply the same techniques discussed in

Appendix A to the second term to account for this term and produce corrected estimators. In particular, the *CV one-step estimator*,

$$\phi_{n,\mathrm{cvos}} := \phi^{\dagger}_{n,\mathrm{cv}} + \frac{1}{K}\sum_{k=1}^{K} P^{1}_{n,k}\{D_{\psi^{0}_{n,k}}(Q^{0}_{n,k}, G^{0}_{n,k})\}. \tag{7}$$

performs an additive correction to $\phi^{\dagger}_{n,\mathrm{cv}}$, moving the second term in the sum to the left-hand-side of equation (6). In situations where the gradient can be expressed as an estimating function, we define the CV estimating equations estimator $\phi_{n,\mathrm{cvee}}$ as the solution in Φ of

$$0 = \frac{1}{K}\sum_{k=1}^{K} P^{1}_{n,k}\{D_{\psi^{0}_{n,k}}(Q^{0}_{n,k}, G^{0}_{n,k}, \Phi)\}. \tag{8}$$

A cross-validated TMLE (CVTMLE) estimator (Zheng and van der Laan, 2011) may be generated by iteratively minimizing a loss function along parametric submodels through components of $Q^{0}_{n,k}$ rather than $Q^{1}_{n,k}$ as in the standard TMLE procedure described in Appendix A. The result is a plug-in estimator of the form (5), but using de-biased estimates $Q^{*}_{n,k}$ of $Q_{0,k}$ for each k Both the estimating equation-based and CVTML estimators can be studied using the expansion (6); however, by construction these estimators satisfy that the problematic second term $\frac{1}{K}\sum_{k=1}^{K} P^{1}_{n,k}\left\{D_{\psi^{0}_{n,k}}(Q^{0}_{n,k}, G^{0}_{n,k})\right\} = o_{\mathrm{p}}(n^{-1/2})$.

We have the following theorem that describes the weak convergence of these three estimators. The proof primarily involves establishing conditions that ensure the asymptotic negligibility of the third and fourth terms of the expansion (6); see the web supplement for expanded discussion.

Theorem 1.

We assume that

(i)
$$\sup_{o \in \mathcal{O}} \sup_{\psi \in \Psi} \left| D(Q_{0,\psi}, G_{0,\psi})(o) \right| < \infty;$$

(ii)
$$P_0\left[\left\{D_{\psi^{0}_{n,k}}\left(Q^{0}_{n,k}, G^{0}_{n,k}\right) - D_{\psi^{0}_{n,k}}\left(Q_{0,k}, G_{0,k}\right)\right\}^{2}\right] = o_{\mathrm{p}}(1) \text{ for each } k;$$

(iii)
$$R_{\psi^{0}_{n,k}}\left(Q^{0}_{n,k}, Q_{0,k}, G^{0}_{n,k}, G_{0,k}\right) = o_{\mathrm{p}}\left(n^{-1/2}\right) \text{ for each } k; \text{ and}$$

(iv) *there exists* ψ^{*} *such that*

$$P_0\left[\left\{D_{\psi_n}\left(Q_{0,\psi_n}, G_{0,\psi_n}\right) - D_{\psi^{*}}\left(Q_{0,\psi^{*}}, G_{0,\psi^{*}}\right)\right\}^{2}\right] = o_{\mathrm{p}}(1).$$

In the case of the CVTML estimator we make these assumptions replacing $Q^0_{n,k}$ *by* $Q^*_{n,k}$. *Then* $n^{1/2}(\phi_{n,cvos} - \phi_{0,cv})$, $n^{1/2}(\phi_{n,cvee} - \phi_{0,cv})$, *and* $n^{1/2}(\phi_{n,cvtmle} - \phi_{0,cv})$ *each converge in distribution to a random variable with a mean-zero Normal distribution with variance* $P_0\{D_{\psi^*}(Q_{0,\psi^*}, G_{0,\psi^*})^2\}$.

Theorem 1 establishes a foundation for statistical inference. Assuming the relevant nuisance estimators satisfy Glivenko-Cantelli conditions (van der Vaart and Wellner, 1996), then $\sigma_n^2 := \frac{1}{K}\sum_{k=1}^{K} P^1_{n,k}\{D^0_{\psi_{n,k}}(Q^0_{n,k}, G^0_{n,k})^2\}$ is a consistent estimate of the asymptotic variance of $n^{1/2}$ times any one of our estimators. We can use this variance estimate to build confidence intervals for $\psi_{0,\mathrm{cv}}$. For example, the interval $\phi_{n,\mathrm{cvtmle}} \pm z_{1-\alpha/2} n^{-1/2} \sigma_n$, where $z_{1-\alpha/2}$ is the $1-\alpha/2$ quantile of a standard Normal distribution will have asymptotic coverage probability no smaller than $1-\alpha$.

5 Examples

We now explicitly apply our framework to two nonlinear prediction metrics. To define these metrics, we require additional notation. For a given $P \in \mathcal{M}$, ψ, and for $y = 0, 1$, we denote by $F_{P,\psi,y}$ the conditional cumulative distribution of $\psi(X)$ given $Y = y$ implied by P; thus, for every $u \in (0,1)$, $F_{P,\psi,y}(u) := \mathrm{pr}_P(\psi(X) \le u | Y = y)$. Similarly, we denote by $F^{-1}_{P,\psi,y}(\rho)$ the ρ-th quantile of the conditional distribution of $\psi(X)$ given $Y = y$ thus, $F^{-1}_{P,\psi,y}(\rho) := \inf\{u : F_{P,\psi,y}(u) \ge \rho\}$. We denote by $\bar{F}_{P,\psi}$ the marginal cumulative distribution of $\psi(X)$, $\bar{F}_{P,\psi}(u) := \mathrm{pr}_P(\psi(X) \le u)$. Finally we define $g_P := \mathrm{pr}_P(Y = 1)$ as the marginal probability of a case. To simplify notation, when possible, we will suppress the indexing of these quantities on P and write e.g., $F_{\psi,y}$ to denote $F_{P,\psi,y}$ for a general $P \in \mathcal{M}$. we additionally introduce the zero subscript as shorthand to denote a quantity evaluated under P_0, e.g., $F_{0,\psi,y} := F_{P_0,\psi,y}$.

5.1 Area under the receiver operating characteristics curve

The AUC is defined for a given as $\psi \in \boldsymbol{\psi}$ as $\Phi_{\psi,\mathrm{AUC}}(P) := \int\{1 - F_{P,\psi,1}(u)\} dF_{P,\psi,0}(u)$. The parameter $\Phi_{\psi,\mathrm{AUC}}$ depends on P through $Q_\psi(P) = (F_{P,\psi,0}, F_{P,\psi,1})$. The efficient influence function of the AUC parameter with respect to $\mathcal{M}$ additionally depends on $G(P) = g_P$. As with the other nuisance parameters, we hence write g when referring to g_P for a general $P \in \mathcal{M}$ and write g_0 when referring to g_{P_0}. We introduce the shorthand for $y = 0, 1$ and a general $P \in \mathcal{M}$, $\tilde{g}_y := y g_P + (1-y)(1 - g_P)$. The efficient influence function of the AUC parameter can be written (LeDell et al., 2015), $D_\psi = D_{\psi,1} + D_{\psi,0}$ where for $y = 0, 1$,

$$D_{\psi,y}(Q_\psi, G_\psi)(O_i) = \frac{(-1)^{Y_i}}{\tilde{g}_{Y_i}} \int_0^1 \{I(\psi(X_i) \le u) - F_{\psi,y}(u)\} dF_{\psi,1-y}(u).$$

We can show through straightforward calculation that the remainder term is

$$R_{\psi}\left(Q_{\psi}, Q_{0,\psi}, G, G_{0,\psi}\right) := \frac{g-g_0}{g}\int_0^1 \{F_{0,\psi,1}(u) - F_{\psi,1}(u)\} dF_{\psi,0}(u)$$
$$+ \frac{g-g_0}{1-g}\int_0^1 \{F_{0,\psi,0}(u) - F_{u,0}(u)\} dF_{\psi,1}(u)$$
$$- \int_0^1 \{F_{0,\psi,0}(u) - F_{\psi,0}(u)\} d(F_{0,\psi,1} - F_{\psi,1})(u).$$

The remainder has a doubly-robust structure: each term involves a cross-product of nuisance parameters under P and P_0. Such terms are often negligible, even in finite samples, since inaccurate estimation of one nuisance parameter may be mitigated by accurate estimation of another. In particular, the first two terms of the remainder are unlikely to contribute substantially since g_0 is easily estimated at $n^{1/2}$-rate. The third term appears more likely to contribute to the finite-sample behavior of the estimator, though it too may not contribute substantially due to the possibility of cancellation of terms in the integrand.

5.1.1 Proposed estimators of AUC—

We consider constructing plug-in estimators of AUC based on two different estimators of $Q_{0,k}$. The first is the empirical cumulative distribution of $\psi^0_{n,k}$. Given $\psi^0_{n,k}$ and $u \in [0,1]$, this estimator is defined as $F^0_{n,k,y}(u) := \mathrm{pr}_{P^0_{n,k}}\left(\psi^0_{n,k}(X) \le u \mid Y = y\right)$. However, this estimator may be insufficient for aggressive algorithms since the same data are used to generate as $\psi^0_{n,k}$ to estimate its conditional cumulative distribution function. Thus, the estimators may not satisfy assumption (iii) of Theorem 1, which could in turn result in biased estimates of prediction metrics in small samples. As an alternative, we propose an estimator of $F_{0,k,y}$ based on nested CV in the training sample. We split the k-th training sample into V additional folds. We denote by $P^0_{n,k,v}$ and $P^1_{n,k,v}$ the empirical distribution of the v-th training and validation samples, respectively, nested in the k-th training sample. We define $\psi_{n,k,v} := \widehat{\Psi}(P^0_{n,k,v})$ to be the prediction algorithm trained using the v-th training fold nested in the k-th training fold. We denote by $\mathscr{V}_{k,v,y}$ the set of indexes of observations that fall in the v-th validation fold nested in the k-th training fold with $Y = y$. For $u \in (0,1)$ and for $y = 0, 1$, we define a CV estimate of the cumulative distribution of

$$\psi^0_{n,k}(X), F^{\mathrm{cv}}_{n,k,y}(u) := \frac{1}{V}\sum_{v=1}^{V} \mathrm{pr}_{P^1_{n,k,v}}\left(\psi_{n,k,v}(X) \le u \middle| Y = y\right)$$
$$= \frac{1}{V}\sum_{v=1}^{V} \frac{1}{|\mathscr{V}_{k,v,y}|} \sum_{i \in \mathscr{V}_{k,v,y}} I\left(\psi_{n,k,v}(X_i) \le u\right),$$

where we take $1/0 = 0$. In the remainder of this subsection, we write the estimate of the conditional cumulative distribution given $Y = y$ as $F_{n,k,y}$, with the understanding that this estimate could be $F^0_{n,k,y}$ or $F^{\mathrm{cv}}_{n,k,y}$. We let $Q^0_{n,k} = (F_{n,k,y} : y = 0, 1)$ and $G^0_{n,k} = g^0_{n,k}$, where $g^0_{n,k} := \mathrm{pr}_{p^0_{n,k}}(Y = 1)$.

Using these definitions, the one step estimator may be constructed directly using equation (7),

$$\phi_{n,\,\mathrm{AUC,\,os}} := \frac{1}{K}\sum_{k=1}^{K}\left[\int_0^1 \left\{1 - F_{n,k,1}(u)\right\} dF_{n,k,0}(u) + P^1_{n,k} D_{\psi^0_{n,k}}(Q^0_{n,k}, G^0_{n,k})\right].$$

The estimating equations estimator can be constructed by rewriting the efficient influence function for a given ψ and a typical observation O_i as

$$D_\psi\left(Q_\psi, G_\psi, \Phi_\psi\right)(O_i) = \frac{I(Y_i = 1)}{g}\left\{F_{\psi,0}(\psi(X_i)) - \Phi_\psi\right\} + \frac{I(Y_i = 0)}{1-g}\left\{1 - F_{\psi,1}(\psi(X_i)) - \Phi_\psi\right\}.$$

The estimating equations estimator $\phi_{n,\,\mathrm{AUC,ee}}$ is defined as the solution in Φ of the equation

$$0 = \frac{1}{K}\sum_{k=1}^{K} P^1_{n,k} D_{\psi^0_{n,k}}(Q^0_{n,k}, G^0_{n,k}, \Phi).$$

Details of the CVTMLE are provided in Appendix B.

5.2 Sensitivity constrained rate of negative prediction

The SCRNP is defined for a given ψ as $\Phi_{\psi,\,\mathrm{SCRNP}}(P) := \bar{F}_{P,\psi}(F^{-1}_{P,\psi,1}(\rho))$. The SCRNP depends on P through nuisance parameter $Q_\psi(P) = (F_{P,\psi,0}, F_{P,\psi,1}, g_P)$. SCRNP is closely related to the positive predictive value of a classifier based on ψ that categorizes an observation x as a case if $\psi(x) > F^{-1}_{\psi,1}(\rho)$ and as a control otherwise. The sensitivity (i.e., probability of correct classification of a true case) of this classifier is $1 - \rho$. The positive predictive value (i.e., the probability of being a true case if classified as such) for a given $P \in \mathcal{M}$ can be written as $(1-\rho)g_P/\{1 - \Phi_{\psi,\,\mathrm{SCRNP}}(P)\}$. Thus, we conclude that large values of SCRNP correspond with large values of positive predictive value for a classifier based on ψ that has sensitivity $1 - \rho$.

Given ψ, the SCRNP parameter depends on P through $Q_\psi(P) = \left(F_{P,\psi,0}, F_{P,\psi,1}, g_P\right)$. For this parameter, no additional nuisance parameter $G_\psi(P)$ appears in the nonparametric efficient influence function at P, so we suppress this notation. We introduce the shorthand $c_{0,\psi} := F^{-1}_{0,\psi,1}(\rho)$ and $c_\psi := F^{-1}_{P,\psi,1}(\rho)$ for a general $P \in \mathcal{M}$. The following theorem establishes the efficient influence function of Φ_{SCRNP} with respect to a nonparametric model.

Theorem 2.—*Suppose $F_{\psi,1}$ and $\bar{F}_\psi$ have densities $f_{\psi,1}$ and $\bar{f}_\psi$ with respect to Lebesgue measure. The efficient influence function of $\Phi_{\psi,\,\mathrm{SCRNP}}$ at $P \in \mathcal{M}$ evaluated on a typical observation O_i is*

$$D(Q_\psi)(O_i) := I(\psi(X_i) \le c_\psi) - \bar{F}_\psi(c_\psi) - \frac{Y_i}{g}\frac{\bar{f}_\psi(c_\psi)}{f_{\psi,1}(c_\psi)}\{I(\psi(X_i) \le c_\psi) - \rho\}.$$

The proof of Theorem 2 is given in Appendix B. The following corollary is implied by equation (2) and Theorem 2. We define

$f'_{0,\psi,1}(z) := \frac{d}{du} f_{0,\psi,1}(u)|_{u=z}$, and $\bar{f}'_{0,\psi}(z) := \frac{d}{du}\bar{f}_{0,\psi}(u)|_{u=z}$.

Corollary 1.—*Suppose that the densities $f_{0,\psi,1}$ and $\bar{f}_{0,\psi}$ are continuously differentiable at $F^{-1}_{0,\psi,1}(\rho)$. Given $\psi \in \boldsymbol{\psi}$, $P \in \mathcal{M}$, we have following linear expansion*

$$\Phi_{\psi,\mathrm{SCRNP}}(Q_\psi) - \Phi_{\psi,\mathrm{SCRNP}}(Q_{0,\psi}) = -P_0 D(Q_\psi) + \sum_{j=1}^{5} R_{j,\psi}(Q_\psi, Q_{0,\psi}), \text{ where}$$

$$R_{1,\psi}(Q_\psi, Q_{0,\psi}) := -\frac{\bar{f}_\psi(c_\psi)}{\bar{f}_\psi(c_\psi)}\left(\frac{g_0 - g}{g}\right)\{F_{0,\psi,1}(c_\psi) - \rho\},$$

$$R_{2,\psi}(Q_\psi, Q_{0,\psi}) := \frac{\bar{f}'_{0,\psi}(c_{1,\psi})}{2}\{c_\psi - c_{0,\psi}\}^2,$$

$$R_{3,\psi}(Q_\psi, Q_{0,\psi}) := \frac{\bar{f}_{0,\psi}(c_{0,\psi}) f'_{0,\psi,1}(c_{2,\psi})}{2 f^3_{0,\psi,1}(c_{2,\psi})}\{c_\psi - c_{0,\psi}\}^2,$$

$$R_{4,\psi}(Q_\psi, Q_{0,\psi}) := \{\frac{\bar{f}_\psi(c_\psi)}{\bar{f}_{1,\psi}(c_\psi)} - \frac{\bar{f}_{0,\psi}(c_{0,\psi})}{\bar{f}_{0,1,\psi}(c_{0,\psi})}\}\{F_{\psi,1}(c_{0,\psi}) - \rho\},$$

$$R_{5,\psi}(Q_\psi, Q_{0,\psi}) := \frac{\bar{f}_\psi(c_\psi)}{f_{1,\psi}(c_\psi)}\{(F_{\psi,1} - F_{0,\psi,1})(c_\psi) - (F_{\psi,1} - F_{0,\psi,1})(c_{0,\psi})\},$$

for some $c_{1,\psi}$ and $c_{2,\psi}$ between c_ψ and $c_{0,\psi}$.

The proof of Corollary 1 is given in Appendix B. We note that R_1, R_4, and R_5 are doubly-robust terms, while R_2 and R_3 involve a squared difference in a conditional quantile implied by P and P_0. Thus, if in a given sample the estimate $c_{n,\psi}$ is a poor approximation of the true conditional quantile $C_{0,\psi}$, R_2 and R_3 will contribute substantially to the behavior of the estimator.

5.2.1 Proposed estimators of SCRNP—As in Section 5.1, we consider empirical estimators of $F_{0,k,1}$ and $F_{0,k,0}$, as well estimators based on V-fold nested CV. We denote by $F_{n,k,1}$ and $F_{n,k,0}$, the chosen estimator of $F_{0,k,1}$ and $F_{0,k,0}$, respectively. Similarly, we can construct estimates $f_{n,k,1}$ and $\bar{f}_{n,k}$ of, respectively, $f_{0,k,1} := f_{0,\psi^0_{n,k},1}$ and $\bar{f}_{0,k} := \bar{f}_{0,\psi^0_{n,k}}$ via kernel regression with CV bandwidth selection. As with the cumulative distribution functions, we can estimate this densities either using the training sample data, or a nested CV approach (described explicitly in the web supplement). Permitting a slight abuse of notation, we define $Q^0_{n,k} := \left(F_{n,k,0}, F_{n,k,1}, f^0_{n,k,1}, \bar{f}^0_{n,k}, g^0_{n,k}\right)$. Using these estimators of the relevant nuisance parameters, we can construct the estimators of the SCRNP. Note that the efficient influence function of $\Phi_{\psi,\text{SCRNP}}$ is linear in $\Phi_{\psi,\text{SCRNP}}$; thus, the one-step estimator and estimating equations estimators are equivalently described by equation (7),

$$\phi_{n,\text{SCRNP,cvos}} = \frac{1}{K}\sum_{k=1}^{K}\left[\left\{g^0_{n,k}\,\rho + \left(1-g^0_{n,k}\right)F_{n,k,0}\left(F^{-1}_{n,k,1}(\rho)\right)\right\} + P^1_{n,k}D_{\psi^0_{n,k}}\left(Q^0_{n,k}\right)\right].$$

A description of the CVTMLE is included in the web supplement.

5.3 Additional examples

The results in Theorem 1 can be applied to many further prediction metrics. For example, consider quantiles of the distribution of absolute error. Given ψ and $P \in \mathcal{M}$, we define $\bar{H}_{P,\psi}$ as the cumulative distribution function of the random variable $|Y - \psi(X)|$ implied by P. The ρ-th quantile is denoted by $\Phi_{\psi,\rho}: \mathcal{M} \to [0,1]$, where $\Phi_{\psi,\rho}(P) := \bar{H}^{-1}_{P,\psi}(\rho)$. For this parameter, we define $Q_\psi(P) = \left(\bar{H}_{P,\psi}, \bar{h}_{P,\psi}\right)$ and $\tilde{c}_\psi = \bar{H}^{-1}_{P,\psi}(\rho)$, where $\bar{h}_{P,\psi}$ is the density of $|Y - \psi(X)|$ implied by P. We can show that the efficient influence function of $\Phi_{\psi,\rho}$ at P evaluated on a typical observation O_i is $D(Q_\psi)(O_i) := \frac{1}{\bar{h}_\psi(\tilde{c}_\psi)}\left\{I\left(|Y_i - \psi(X_i)| \le \tilde{c}_\psi\right) - \rho\right\}$.

Because of the similarity between this influence function and that of the SCRNP parameter, our estimators would require only minor modifications to be applied to this parameter. Theorem 1 also applies to estimation of time-varying prediction metrics such as time-varying AUC in survival analysis settings (Heagerty et al., 2000). In these cases, if dependent censoring is present, then we typically must obtain an estimate of the covariate-conditional censoring distribution to generate a consistent estimator of time-varying AUC. Estimators based on Theorem 1 allow for estimation of the (potentially high-dimensional) censoring distribution in the training sample, which may lead to improved small-sample performance.

6 Simulation

We evaluated our proposed estimators of AUC and SCRNP by simulation. We drew X from a ten-dimensional Normal distribution with mean zero and identity covariance matrix. Given X the outcome Y was generated as from a Bernoulli distribution with case probability given by a logistic linear function of X, $\text{logit}\left\{\text{pr}_{P_0}(Y=1|X)\right\} = 0.25\text{X}_\text{I} + 0.125\text{X}_2X_3 - 0.5\text{X}_4$, where

$\text{logit}(x) = \log\{x/(1-x)\}$. We evaluated two prediction algorithms, logistic regression and random forest, chosen to represent less and more aggressive algorithms, respectively. We considered four sample sizes, $n \in \{50, 100, 250, 500\}$, and for each sample size, we randomly generated 1,000 data sets to evaluate our estimators.

For each data set, we computed our proposed estimators and standard K-fold CV estimators with $K = 5$,10,20, and 40. For each of our proposed estimators, we used both the NPMLE and nested CV-based estimators of nuisance parameters. For our three proposed estimators that were based on nested cross-validation, we used $V = 5$. Whenever CV was used, we stratified by Y to ensure approximately equal numbers of cases in each validation fold. Nevertheless, in the setting with $n = 50$ and $K = 40$ many data sets had at least some validation samples that contained observations all with $Y = 1$ or $Y = 0$. In these cases, the standard CV estimators are not well defined and are thus omitted from results.

In addition to the proposed and standard K-fold CV-based estimators, we computed two bootstrap-corrected estimates, each based on 500 bootstrap re-samples. The first is described in the introduction. The second is the 0.632 corrected bootstrap described in Friedman et al. (2001) adapted to the current problems (full description in web supplement). We also computed the single sample split estimator using 70% of the data to generate the prediction function and 30% to estimate the performance metric. Finally, for study of AUC, we additionally computed the leave-pair-out CV estimator (Airola et al., 2011), which has previously been shown to outperform K-fold CV estimators (Smith et al., 2014). In this approach, each possible pair of one case and one control are left out and the prediction algorithm is developed using the remaining observations. The estimate of AUC is the proportion of the possible case/control pairs for which the case had higher predicted risk than control.

We evaluated the estimates of the performance metric relative to the true value of the performance metric for the prediction algorithm returned by a given procedure. For all but the single sample split approach, the prediction algorithm returned is that developed using the entire data set. For the single sample split, it is the prediction algorithm developed using 70% of the data. The true value of the prediction metric was computed numerically by evaluating the metric on an independent test set of 100,000 observations. Note that because the true prediction metric is defined with respect to each data set separately, it is itself a random variable. Thus, we also present summary statistics of the true performance in our results. We judged our estimates on the absolute value of their bias as a percent of the true target, their coefficient of variation (defined as Monte Carlo standard deviation of the estimates divided by the Monte Carlo mean of the estimates), and their mean squared-error (MSE). We present the mean squared-error of estimators as relative to the standard five-fold CV estimator.

6.1 AUC results

The average true values (interquartile ranges) of the AUC for the logistic regression fit using the entire data set across all simulations were 0.57 (0.55,0.60), 0.59 (0.58,0.61), 0.62 (0.61,0.63) and 0.63 (0.63,0.64) for $n = 50$,100,250, and 500, respectively. Not surprisingly, the true values of the AUC for logistic regression fit to 70% of the data were lower,

demonstrating one drawback of the single sample split approach (web supplement). The bias of each of the standard and novel K-fold CV-based estimators decreased as K increased (Figure 1, top row). The bias of the standard bootstrap estimator tended to be slightly larger, and the bias of the 0.632 bootstrap and leave-pairs-out tended to be slightly smaller. The variability of the novel and standard K-fold CV estimators was relatively unaffected by the number of folds (middle row). The standard bootstrap had the lowest variability across sample sizes and the sample split had the highest variability (middle row). In terms of MSE, the standard bootstrap performed marginally better than other estimators (bottom row). The novel K-fold CV estimators tended to perform better in terms of MSE than standard K-fold estimators. The best choice of K for the novel estimators differed depending on sample size, with small K performing better at $n = 50$ and large K better at $n = 500$. However, for each choice of K, our proposed estimators outperformed the corresponding standard CV estimator.

The average true values (interquartile ranges) of the AUC for the random forest fit using the entire data set across all simulations were 0.56 (0.54,0.68), 0.57 (0.55,0.59), 0.59 (0.58,0.60) and 0.60 (0.59,0.61) for $n = 50,100,250$, and 500, respectively. Again, we found that the true value of the AUC for the random forest fit to only 70% of the data to be lower (web supplement). As with logistic regression, we found that the bias of the standard and novel K-fold-CV-based estimators decreased as K increased (Figure 2, top row). However, both bootstrap estimators were strongly biased at all sample sizes. The leave-pairs-out estimator had low bias in each setting. In terms of MSE, the novel five-fold CV estimating equations estimator performed best at each sample size, offering between 10% and 18% improvement in MSE relative to the standard five-fold CV estimator. Again we found that for each choice of K our proposed estimators performed at least as well as the corresponding standard CV estimator.

Overall, we conclude that the bootstrap corrected estimator had the best performance for logistic regression in terms of MSE, but the worst for random forests. The single sample splitting approach performed poorly both in terms of the true performance of the prediction algorithm and in terms of the variability of estimated performance metric. Our proposed estimators performed well both for logistic regression and random forests offering better performance than standard CV estimators and comparable or slightly better performance than the leave-pairs-out CV estimator. We found that confidence intervals for the novel K-fold-CV-based estimators performed adequately, with coverage probabilities > 80% at all sample sizes and nearing nominal level in larger samples. larger samples.

6.2 SCRNP results

The average true values (interquartile ranges) of SCRNP with 95% sensitivity constraint based on a logistic regression fit to the entire data set across all simulations were 0 067 (0 061,0.074), 0.073 (0.067,0.78), 0.081 (0.078,0.085), and 0.085 (0.084,0.088) for $n = 50,100,250$, and 500, respectively. We found that bias of each of the standard K-fold-CV-based estimator was substantial (often, > 100% of the true value) in small samples (Figures 3), and increased with K On the other hand, the novel CVTML estimators tended to having decreasing bias as K increased. These estimators had smaller bias than the proposed CV

one-step estimator in small samples, and comparable or better bias than the bootstrap-based estimators. The CV one-step estimator had large variability in small samples (middle row) due to unstable estimates of the density of the prediction algorithm used in the one-step correction. Owing to its nature as a plug-in estimator, the CVTMLE had more stable performance. The CVTMLE had > 90% improvement over the standard five-fold estimator in terms of mean squared-error at the smallest sample size and 24% improvement at the largest sample size. The 0.632 bootstrap had strong performance in small samples, but poor performance in larger samples. We conjecture that this is because this estimator mimics approximately two-fold cross-validation. Thus, in small samples, a stable estimate of the conditional quantile is obtained, but in large samples the estimator suffers by only fitting prediction algorithms using half the data. Similar to the AUC simulations, we found that for each K, the newly proposed estimators outperformed standard CV estimators.

The average true values (interquartile ranges) of SCRNP with 95% sensitivity constraint based on a random forest fit to the entire data set across all simulations were 0 062 (0.057,0.067) 0.065 (0.060,0.069), 0.069 (0.066,0.073) and 0072 (0.069,0.076) for n = 50,100,250 and 500, respectively. Results for this setting (Figure 4) were similar to those for logistic regression. However, in this setting, as expected, we find that the standard bootstrap fails to give unbiased estimates of the target parameter. The relative performance of the 0.632-corrected bootstrap is somewhat erratic with worse performance in small and large samples than moderate sized samples. Our proposed CVTMLE performed well overall, offering > 90% improvement in the smallest samples and 35% improvement in the largest samples. We found that confidence intervals for the novel K-fold-CV-based estimators performed adequately and often far superior to the standard K-fold approach. Overall, we conclude that for estimating the SCRNP, the CVTML estimator offers drastic improvements over standard K-fold-CV-based estimators and bootstrap-based estimators.

6.3 Additional simulations

We repeated the above simulations using different choices of V. Full results for $V = 39$, and more limited results for other choices of V are included in the supplement. Overall, we found that using fewer nested CV folds tended to yield the best results. We also repeated the SCRNP simulation enforcing a 65% and 80% sensitivity constraints. Results were largely similar though the magnitude of the benefits of CVTMLE decreased with decreasing sensitivity constraints. Nevertheless, even for the least stringent 65% constraint, CVTMLE still offered up to a 25% improvement in MSE. We evaluated the performance of standard error estimators and confidence intervals for the K-fold-CV-based estimators. We found that the standard error estimators tended to underestimate the true standard error in the smallest sample sizes, which led to slight under-coverage of confidence intervals. With increasing sample size, the standard error estimators were seen to converge to the true standard error and confidence interval coverage approached nominal level.

We additionally evaluated the utility of our estimators in selecting tuning parameters for learning algorithms. In particular, we performed a simulation comparing the generalization error of an elastic net algorithm (Zou and Hastie, 2005) with tuning parameters selected using standard CV estimators compared to our estimators. We found that using our

estimators resulted in modest gains in generalization error. We leave to future research a more encompassing study of this topic.

7 Data Analysis

We analyzed seven publicly available data sets (Table 1, Dheeru and Karra Taniskidou (2017)). For each, we randomly sampled $n \in \{50, 100, 250, 500\}$ observations from the full N observations. These n observations constituted the analysis set, while the remaining $N - n$ observations constituted a hold-out set. Analysis sets with fewer than five cases were discarded and sampling was repeated until there were one hundred unique analysis sets for each data set. The analysis data were used to estimate the AUC and SCRNP (for classifier with 95% sensitivity) for two algorithms. The first used extreme gradient tree boosting (Chen and Guestrin, 2016) based on 500 trees of maximum depth four, a minimum of two observations per tree node, and shrinkage factor of 0.1. The second algorithm used random forests with tuning parameters set to their recommended values (Breiman, 2001). The "true" value of the performance metric was estimated by the computing NPMLE of the metric on the held-out data. As with the simulation, this metric was computed with respect to the algorithm fit to the full analysis data set for all but the single sample splitting approach, where it was computed with respect to the algorithm fit using 70% of the data. We evaluated the same estimators using the same criterion as in the simulation study. However, we omitted the standard bootstrap estimator, as the simulation demonstrated it was not appropriate for machine learning algorithms. We present results for the estimators aggregated over all data sets. In the web supplement, we show results for individual data sets and for a larger choice of nested CV folds V.

7.1 AUC results

Average "true" values of AUC ranged from 0.62 (drugs data set, XGBOOST, $n = 50$) to 0.95 (cardio data set, random forest, $n = 500$). A plot of these values is included in the web supplement. The standard 10-fold CV-based estimators of the AUC of XGBOOST performed best at the smallest sample sizes (Figure 5, top row). At the larger sample sizes our proposed estimators had lower MSE than the standard K-fold-CV-based estimators and MSE about the same as leave-pairs-out CV. For AUC of the random forest, the leave-pairs-out CV estimator performed well at all sample sizes. Our estimators provided comparable performance for all but the smallest sample size. The 0.632-corrected bootstrap had excellent performance in small samples, but poor performance in larger samples. These results were consistent across all data sets (web supplement).

7.2 SCRNP results

The "true" values of SCRNP for a classifier with 95% sensitivity averaged over the one hundred replicates ranged from 0.09 (default data set, XGBOOST, $n = 50$) to 0.70 (cardio data set, random forest, $n = 500$). A plot of all the true values is available in the web supplement. The CVTMLE performed best at the smallest sample sizes for both XGBOOST and random forest (Figure 6, left columns). At larger sample sizes, we found the performance approximately equivalent with that of the standard five-fold CV estimator. This trend was consistent across all data sets (web supplement).

8 Discussion

Our analysis demonstrates the utility of using exact second-order expansions for studying cross-validated prediction metrics. Not only do these expansions highlight potential shortcomings of standard cross-validation-based estimators, they also naturally suggest pathways for developing estimators that have desirable small- and large-sample properties. In particular, our analysis highlights that the second-order remainder, which may be overlooked when considering estimation in large samples, often plays an important role in small-sample estimator performance. Our framework allows one to utilize more data to estimate key nuisance parameters, which can generally be expected to result in more stable estimates of relevant nuisance parameters and thereby better control of the second-order remainder. Nevertheless, one must take care that the training data are appropriately used to estimate the relevant nuisance quantities. Our nested cross-validation-based nuisance estimators provide one strategy for generating nuisance estimators in training data, but other strategies may prove fruitful as well.

Though we have focused on a limited set of learning algorithms, our theory applies generally to any learning approach. However, our numerical studies suggest that the optimal choice of K varies depending on the learning algorithm, the chosen metric and estimator thereof. To achieve optimal performance, we suggest that several options be evaluated in practice, for example, using simulations. Given the added computational burden of these simulations, it is unlikely that this will become standard data analysis practice. Instead, researchers will likely continue to pre-specify a single, fixed K, such as 10 or 20. Therefore, it is important to highlight that in our simulations, our estimators almost always performed at least as well as the standard cross-validation estimators for these choices of K. In the future, we may provide a more extensive evaluation of particular learning algorithms to establish more informed recommendations for general use.

Our simulation results demonstrated that in modest-sized samples, there is no benefit to the single sample splitting approach. While this approach is considered standard practice when considering analysis of large data sets, in smaller data sets, it suffers considerable drawbacks: the resultant prediction functions generally have worse performance than those fit using the the full data set and the resultant estimates of performance are significantly worse than those based on alternative approaches. Our simulation results also suggest that bootstrap approaches are not often unreliable. For these reasons, K-fold CV-based approaches should preferred. In addition to improved estimation, these estimators additionally afford closed-form inference, which is not available for bootstrap or leave-pair-out approaches. Moreover, our simulation results suggest that, contrary to the conclusions of previous works (e.g., Friedman et al. (2001)), K-fold CV estimators may reasonably be used to infer the generalization error of an algorithm. Amongst K-fold approaches, our estimators provided improved estimation for the two examples we considered.

An R package, nipred, that implements our proposed methods is available in the web supplemental material and on the Comprehensive R Archive Network (https://cran.r-project.org/).

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Appendix A.: Strategies to correct for bias

The one-step correction uses $\phi_{n,\mathrm{os}} := \phi_{n,\mathrm{cv}} + P^1_{n,k} D_{\psi^0_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}\right)$, as an estimate of $\phi_{0,\mathrm{cv}}$ (Ibragimov and Has'minskii, 1981; Pfanzagl, 1982). The estimating equations approach requires that, given ψ and P, the gradient of Φ_ψ can be expressed as a function of $\Phi_\psi(P)$. If so, then we may define the estimating equations estimator $\phi_{n,\mathrm{ee}}$ as the solution in ϕ of $0 = \frac{1}{K}\sum_{k=1}^{K} P^1_{n,k}\left\{D_{\psi^0_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}, \phi\right)\right\}$. The targeted minimum loss-based estimation (TMLE) approach involves a two-step process for generating estimators of $Q_{0,k}$ and $G_{0,k}$ that ensure $P^1_{n,k} D_{\psi^0_{n,k}}\left(Q^1_{n,k}, G^1_{n,k}\right) = o_{\mathrm{p}}\left(n^{-1/2}\right)$ (van der Laan and G Rubin, 2006). First, initial estimates $Q^{\mathrm{init}}_{n,k}$ of $Q_{0,k}$, and $G^1_{n,\psi}$ of $G_{0,\psi}$ are generated. We denote by $Q_{k,\overline{m}}$ the $m-1$ components of Q_k other than $Q_{k,m}$. Given $Q_{k,\overline{m}}$ and G, for $m = 1, \ldots, M, k = 1, \ldots, K$ we define a valid loss function $\left(Q_{k,m}, O_i\right) \mapsto \mathbb{L}\left(Q_{k,m}, O_i \mid Q_{k,\overline{m}}, G\right)$. a loss function is valid for estimation of $Q_{0,k,m}$ If $Q_{0,k,m} = \mathrm{argmin}_{q \in \mathbb{Q}_{k,m}} \mathrm{E}_{P_0}\left\{\mathbb{L}\left(q, O \mid Q_{0,k,\overline{m}}, G_{0,\psi}\right)\right\}$. If each loss function is valid for its respective target, then the sum loss $\mathbb{L}_k(\cdot) = \sum_{m=1}^{M} \mathbb{L}\left(\cdot \mid Q_{0,k,\overline{m}}, G\right)$ is valid for $Q_{0,\ k}$. Next, permitting a slight abuse of notation, we set $Q^{i=1}_{n,k} = Q^{\mathrm{init}}_{n,k}$, and recursively define a univariate parametric submodel $\mathcal{Q}^i \subseteq \mathcal{M}$ through the i-th estimate $\left(Q^i_{n,k} : k\right), \mathcal{Q}^i = \left\{P \in \mathcal{M} : Q_{\psi^0_{n,k}}(P) = Q^i_{n,k,\epsilon}, k = 1, \ldots, K, \epsilon \in \mathrm{Y}\right\}$, where $Q^i_{n,k,\epsilon}$ is such that $Q^i_{n,k,0} = Q^i_{n,k}$. The submodel may additionally be indexed by $G^1_{n,\psi}$; however, we omit this additional notation. We assume that for each i, $\mathcal{Q}^i$ and $\mathbb{L}$ satisfy $\frac{d}{d\epsilon}\mathbb{L}\left(Q_{n,k,\epsilon}, O_i \mid Q^i_{n,k,\overline{m}}, G^1_{n,k}\right)\Big|_{\epsilon=0} \propto D_{\psi^0_{n,k}}\left(Q^i_{n,k}, G^1_{n,k}\right)(O_i)$. We define the MLE, $\epsilon^i_n := \mathrm{argmin}_{\epsilon \in \mathrm{Y}} \sum_{k=1}^{M}\sum_{m=1}^{M} \mathrm{E}_{P^1_{n,k}}\left\{\mathbb{L}\left(Q^i_{n,k,\epsilon}, O \mid Q^i_{n,k,\overline{m}}, G^1_{n,k}\right)\right\}$, and set $Q^{i+1}_{n,k} = Q_{n,k,\epsilon^i_n}$. This process continues until $\sum_{k=1}^{K} P^1_{n,k} D_{\psi^0_{n,k}}\left(Q^i_{n,k}, G\right) \leq s_n$, where s_n is a stopping criteria chosen such that $S_n = o_{\mathrm{p}}\left(n^{-1/2}\right)$. Suppose this convergence occurs after j iterations. We define $Q^*_{n,k} := Q^j_{n,k}$. We can generally prove that, under regularity conditions, relevant properties (e.g., the convergence rate) of the initial estimator $Q^{\mathrm{init}}_{n,k}$ are inherited by the targeted estimator $Q^*_{n,k}$ (van der Laan, 2017). The TMLE of $\phi_{0,\mathrm{cv}}$ is $\phi_{n,\mathrm{tmle}} := \frac{1}{K}\sum_{k=1}^{K} \Phi_{\psi^0_{n,k}}\left(Q^*_{n,k}\right)$.

Appendix B.: CVTMLE for cross-validated AUC

For the CVTMLE, we require an appropriate submodel and loss function. Given $y = 0, 1$ and an estimate $F^{\circ}_{n,k,y}$ of $F_{0,k,y}$ for $k = 1, \ldots, K$, we define a submodel,

$$\mathscr{F}_y := \left\{ P \in \mathscr{M} : \text{logit}(F_{P,\psi^0_{n,k},y}) = \text{logit}(F^\circ_{n,k,y}) + \epsilon \text{ for } k = 1,\ldots,K, y = 0,1, \text{ and } \epsilon \in \mathbb{R} \right\}.$$

We define the integrated negative log-likelihood loss as

$$\mathbb{L}(F_{k,y}, O_i \mid F_{k,1-y}, g) := -\frac{I(Y_i = y)}{\tilde{g}_{Y_i}} \int_0^1 \log\left[F_{k,y}(u)^{I(\psi^0_{n,k}(X_i) \le u)} \{1 - F_{k,y}(u)\}^{I(\psi^0_{n,k}(X_i) > u)} \right] dF_{k,1-y}(u).$$

One can confirm that $F_{0,\ k,y}$ minimizes $\mathrm{E}_{P_0}\{\mathbb{L}(f, O \mid F_{k,1-y}, g)\}$ over all f in the class of monotone increasing functions on [0,1] for any choice of $F_{k,1-y}$ and g and for $y = 0, 1$. Thus, $\mathbb{L}$ is a valid loss function for $F_{0,k,y}$. Moreover, defining $F_{n,k,y,\epsilon_y} := \text{expit}\{\text{logit}(F^\circ_{n,k,y}) + \epsilon_y\}$ for $k = 1,\ldots,K,\ y = 0,1,$ and $\epsilon_y \in \mathbb{R}$, we can also confirm that

$$\frac{d}{d\epsilon_y}\mathbb{L}(F_{n,k,y,\epsilon_y}, O_i \mid F_{k,1-y}, g)\Big|_{\epsilon_y = 0} = \frac{I(Y_i = y)}{\tilde{g}_{Y_i}} \int_0^1 \{I(\psi(X_i) \le u) - F^\circ_{n,k,y}(u)\} dF_{k,1-y}(u).$$

Thus, we can conclude that this submodel/loss combination is valid for use in CVTMLE. The CVTMLE procedure proceeds as described for the general problem in Appendix A. We start with initial estimate $F^{i=1}_{n,k,y} := F_{n,k,y}$. Given current estimates $F^i_{n,k,y}, y = 0, 1$, we proceed by first updating the estimates of $F_{0,k,0}$ by finding

$$\epsilon_{n,0} := \underset{\epsilon_0 \in \mathbb{R}}{\text{argmin}} \sum_{k=1}^K \mathrm{E}_{P^1_{n,k}}\left\{\mathbb{L}\left(F^i_{n,k,0,\epsilon_0}, O \mid F_{k,1,n}, g^1_n\right)\right\},$$

where $g^1_n = \text{pr}_{P^1_{n,k}}(Y = 1)$. We define the updated estimate $F^{i+1}_{n,k,0} = \text{expit}\left[\text{logit}\left\{F^i_{n,k,0}\right\} + \epsilon_{n,0}\right]$. Next, we update the estimates of $F_{0,k,1}$ by finding

$$\epsilon_{n,1} := \underset{\epsilon_1 \in \mathbb{R}}{\text{argmin}} \sum_{k=1}^K \mathrm{E}_{P^1_{n,k}}\left\{\mathbb{L}\left(F^i_{n,k1,\epsilon_1}, O \mid F^{i+1}_{k,1,n}, g^1_n\right)\right\},$$

and defining the updated estimate $F^{i+1}_{n,k,1} = \text{expit}\left[\text{logit}\left\{F^i_{n,k,1}\right\} + \epsilon_{n,1}\right]$. We continue this iterative updating process until $K^{-1}\sum_{k=1}^K P^1_{n,k} D_{\psi^0_{n,k}}\left(Q^i_{n,k}, G^0_{n,k}\right)$ is smaller than $n^{-1/2}$, where $Q^i_{n,k} = \left(F^i_{n,k,y}, y = 0, 1\right)$. In our simulations, this criterion was often satisfied after a single iteration. We denote by $F^*_{n,k,y}$ the estimates after their final update. The CVTMLE of cross-validated AUC is

$$\phi_{n,\mathrm{AUC,cvtmle}} := \frac{1}{K}\sum_{k=1}^{K}\int_0^1 \left\{1 - F^*_{n,k,1}(u)\right\} dF^*_{n,k,0}(u).$$

Appendix C.: CVTMLE for cross-validated SCRNP

To construct the CVTMLE, we first rewrite the efficient influence function as

$$D(Q_{0,\psi}) = D_{\psi|Y}(Q_{0,\psi}) + D_{\psi,Y}(Q_{0,\psi}), \text{ where}$$

$$D_{\psi,Y}(Q_{0,\psi})(O_i) = F_{0,\psi,Y_i}\left(F^{-1}_{0,\psi,1}(\rho)\right) - \Phi_{\psi,\mathrm{SCRNP}}(Q_{0,\psi}) \text{ and}$$

$$D_{\psi|Y}\left(Q_{0,\psi}\right)(O_i) = \left(1 - \frac{Y_i}{g_0}\frac{\bar{f}_{0,\psi}\left(F^{-1}_{0,\psi,1}(\rho)\right)}{f_{0,\psi,1}\left(F^{-1}_{0,\psi,1}(\rho)\right)}\right)\left\{I\left(\psi(X_i) \le F^{-1}_{0,\psi,1}(\rho)\right) - F_{0,\psi,Y_i}\left(F^{-1}_{0,\psi,1}(\rho)\right)\right\}.$$

Given estimates $F^0_{n,k,y}$ of $F_{0,k,y}$ for $y = 0, 1$ and $k = 1, \ldots, K$, we define the submodel

$$\mathcal{F}_{F^{-1}_1}(\rho) := \left\{P \in \mathcal{M} : \mathrm{logit}\left\{F_{P,\psi^0_{n,k},y}\left(F^{-1}_{k,1}(\rho)\right)\right\} = \mathrm{logit}\left\{F_{n,k,y}\left(F^{-1}_{k,1}(\rho)\right)\right\} + \epsilon\right.$$

for $k = 1, \ldots, K, y = 0, 1,$ and $\epsilon \in \mathbb{R}\}$, with $F^{-1}_{k,1}(\rho)$ considered fixed for each k. We define a weighted negative log-likelihood loss function for $F_{0,\psi^0_{n,k},y}\left(F^{-1}_{k,1}(\rho)\right)$ for a given $F^{-1}_{k,1}(\rho)$. Specifically, defining

$$w\left(Y_i \middle| g, \bar{f}_k, f_k, F^{-1}_{k,1}(\rho)\right) := 1 - \frac{Y_i}{g}\frac{\bar{f}_{0,\psi}\left(F^{-1}_{k,1}(\rho)\right)}{f_{0,\psi,1}\left(F^{-1}_{k,1}(\rho)\right)},$$

the loss function may be written as $\left(F_{k,y}, O_i\right) \mapsto \mathbb{L}\left(F_{k,y}, O_i | g, \bar{f}_k, f_{k,1}, F^{-1}_{k,1}(\rho)\right)$, where

$$\mathbb{L}\left(F_{k,y}, O_i \middle| F^{-1}_{k,1}(\rho), g, \bar{f}_k, f_{k,1}\right) := -w\left(Y_i \middle| g, \bar{f}_k, f_{k,1}, F^{-1}_{k,1}(\rho)\right)$$
$$\times \log\left[F_{k,y}\left(F^{-1}_{k,1}(\rho)\right)^{I\left(\psi^0_{n,k}(X_i) \le F^{-1}_{k,1}(\rho)\right)}\left\{1 - F_{k,y}\left(F^{-1}_{k,1}(\rho)\right)\right\}^{1 - I\left(\psi^0_{n,k}(X_i) \le F^1_k(\rho)\right)}\right].$$

One can confirm that $F_{0,k,y}\left(F^{-1}_{k,1}(\rho)\right)$ minimizes $E_{P_0}\left\{\mathbb{L}\left(f, O | F^{-1}_{k,1}(\rho), g, \bar{f}_k, f_{k,1}\right)\right\}$ over all $f \in (0,1)$. Thus, $\mathbb{L}$ is a valid loss function for $F_{0,k,y}\left(F^{-1}_{k,1}(\rho)\right)$. Moreover, defining $F_{n,k,y,\epsilon}(u) := \mathrm{expit}\left\{\mathrm{logit}\left(F_{n,k,y}(u)\right) + \epsilon\right\}$ for $k = 1, \ldots, K, y = 0, 1, \epsilon \in \mathbb{R}$, and a fixed $u \in (0,1)$, we can also confirm that

$$\frac{d}{d\epsilon}\sum_{y=0}^{1}\mathbb{L}\left(F_{n,k,y,\epsilon},O_i\middle|F^{-1}_{n,k,1}(\rho),g^0_{n,k},\bar{f}_{n,k},f_{n,k,1}\right)\Big|_{\epsilon=0}=D_{\psi^0_{n,k}}\middle|Y\left(Q^0_{n,k}\right).$$

The CVTMLE procedure proceeds as described for the general problem in Appendix A. We start with initial estimate $F^{i=1}_{n,k,y}:=F^0_{n,k,y}$ of $F_{0,k,y}, y=0,1$ and estimates $f^0_{n,k,1}$ and $\bar{f}^0_{n,k}$ of, respectively, $f_{0,k,1}$ and $\bar{f}_{0,k}$. We update $F^i_{n,k,y}$, $y=0,1$, by computing

$$\epsilon_n:=\underset{\epsilon\in\mathbb{R}}{\operatorname{argmin}}\sum_{k=1}^{K}\sum_{y=0,1}\mathrm{E}P^1_{n,k}\left\{\mathbb{L}\left(F^i_{n,k,y},O_i\middle|F^{-1,i}_{n,k,1}(\rho),g^0_{n,k},\bar{f}_{n,k},f_{n,k,1}\right)\right\},$$

and defining the updated estimate $F^{i+1}_{n,k,y}=\mathrm{expit}\left[\mathrm{logit}\left\{F^i_{n,k,y}\right\}+\epsilon_n\right]$. We update the estimate of $F^{-1}_{0,k,1}(\rho)$ according to $F^{i+1}_{n,k,1}$ and repeat the updating procedure using the new estimates. We continue this iterative updating process until $K^{-1}\sum_{k=1}^{K}P^1_{n,k}D_{\psi^0_{n,k}}\left(Q^i_{n,k},G^0_{n,k}\right)$ is smaller than $n^{-1/2}$, where $Q^i_{n,k}=\left(F^i_{n,k,0},F^i_{n,k,1},f^0_{n,k,1},\bar{f}^0_{n,k}\right)$. In our simulations, this criterion was often satisfied after a single iteration. We denote by $F^*_{n,k,y}$ the estimates after their final update. The CVTMLE of cross-validated SCRNP is

$$\phi_{n,\mathrm{SCRNP,cvtmle}}:=\frac{1}{K}\sum_{k=1}^{K}\left\{g^0_{n,k}\rho+\left(1-g^0_{n,k}\right)F^*_{n,k,0}\left(F^{*\,-1}_{n,k,1}(\rho)\right)\right\}.$$

References

Aeberhard S, Coomans D, and De Vel O. (1992). Comparison of classifiers in high dimensional settings. Dept. Math. Statist., James Cook Univ., North Queensland, Australia, Tech. Rep, 92:02.

Airola A, Pahikkala T, Waegeman W, De Baets B, and Salakoski T. (2011). An experimental comparison of cross-validation techniques for estimating the area under the ROC curve. Computational Statistics & Data Analysis, 55(4):1828–1844.

Austin PC, van Klaveren D, Vergouwe Y, Nieboer D, Lee DS, and Steyerberg EW (2017). Validation of prediction models: examining temporal and geographic stability of baseline risk and estimated covariate effects. Diagnostic and Prognostic Research, 1(1):12. [PubMed: 29350215]

Ayres-de Campos D, Bernardes J, Garrido A, Marques-de Sa J, and Pereira-Leite L. (2000). SisPorto 2.0: a program for automated analysis of cardiotocograms. Journal of Maternal-Fetal Medicine, 9(5):311–318. [PubMed: 11132590]

Benkeser D, Ju C, Lendle S, and van der Laan M. (2017). Online cross-validation-based ensemble learning. Statistics in Medicine, 37(2):249–260. [PubMed: 28474419]

Bock R, Chilingarian A, Gaug M, Hakl F, Hengstebeck T, Jiřina M, Klaschka J, Kotrč E, Savickỳ P, Towers S, et al. (2004). Methods for multidimensional event classification: a case study using images from a Cherenkov gamma-ray telescope. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 516(2–3):511–528.

Breiman L. (2001). Random forests. Machine Learning, 45(1):5–32.

Chen T. and Guestrin C. (2016). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 785–794. ACM.

Dheeru D. and Karra Taniskidou E. (2017). UCI machine learning repository.

Fehrman E, Muhammad AK, Mirkes EM, Egan V, and Gorban AN (2017). The five factor model of personality and evaluation of drug consumption risk. In Data Science, pages 231–242. Springer.

Friedman J, Hastie T, and Tibshirani R. (2001). The elements of statistical learning, volume 1. Springer Series in Statistics New York, NY, USA.

Glümer C, Vistisen D, Borch-Johnsen A, and Colagiuri S. (2006). Risk scores for type 2 diabetes can be applied in some populations but not all. Diabetes Care, 29(2):410–414. [PubMed: 16443896]

Harrell FE, Lee AL, and Mark DB (1996). Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. Statistics in medicine, 15(4):361–387. [PubMed: 8668867]

Heagerty PJ, Lumley T, and Pepe MS (2000). Time-dependent roc curves for censored survival data and a diagnostic marker. Biometrics, 56(2):337–344. [PubMed: 10877287]

Hubbard AE, Kherad-Pajouh S, and van der Laan MJ (2016). Statistical inference for data adaptive target parameters. The International Journal of Biostatistics, 12(1):3–19. [PubMed: 27227715]

Ibragimov I. and Has'minskii R. (1981). Statistical estimation – asymptotic theory. Springer-Verlag Science & Business Media.

Kandasamy A, Krishnamurthy A, Poczos B, Wasserman L, et al. (2015). Nonparametric von Mises estimators for entropies, divergences and mutual informations. In Advances in Neural Information Processing Systems, pages 397–405.

Kohavi R. (1996). Scaling up the accuracy of naive-Bayes classifiers: A decision-tree hybrid. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, volume 96, pages 202–207.

LeDell E, Petersen M, and van der Laan MJ (2015). Computationally efficient confidence intervals for cross-validated area under the ROC curve estimates. Electronic Journal of Statistics, 9(1):1583. [PubMed: 26279737]

Luedtke AR and van der Laan MJ (2016). Super-learning of an optimal dynamic treatment rule. The International Journal of Biostatistics, 12(1):305–332. [PubMed: 27227726]

Moons KG, Kengne AP, Grobbee DE, Royston P, Vergouwe Y, Altman DG, and Woodward M. (2012). Risk prediction models: II. external validation, model updating, and impact assessment. Heart, 98(9):691–698. [PubMed: 22397946]

Moro S, Cortez P, and Rita P. (2014). A data-driven approach to predict the success of bank telemarketing. Decision Support Systems, 62:22–31.

Pfanzagl J. (1982). Contributions to a general asymptotic statistical theory. Springer-Verlag New York.

Smith GC, Seaman SR, Wood AM, Royston P, and White IR (2014). Correcting for optimistic prediction in small data sets. American Journal of Epidemiology, 180(3):318–324. [PubMed: 24966219]

Steyerberg EW, Harrell FE, Borsboom GJ, Eijkemans M, Vergouwe Y, and Habbema JDF (2001). Internal validation of predictive models: efficiency of some procedures for logistic regression analysis. Journal of Clinical Epidemiology, 54(8):774–781. [PubMed: 11470385]

Steyerberg EW, Vickers AJ, Cook NR, Gerds T, Gonen M, Obuchowski N, Pencina MJ, and Kattan MW (2010). Assessing the performance of prediction models: a framework for some traditional and novel measures. Epidemiology, 21(1):128. [PubMed: 20010215]

van der Laan MJ (2017). A generally efficient targeted minimum loss based estimator based on the highly adaptive lasso. The International Journal of Biostatistics, 13(2).

van der Laan MJ and Robins JM (2003). Unified methods for censored longitudinal data and causality. Springer Science & Business Media.

van der Laan MJ and Rubin D. (2006). Targeted maximum likelihood learning. The International Journal of Biostatistics, 2(1):1–40.

van der Vaart AW (2000). Asymptotic statistics. Cambridge University Press.

van der Vaart AW and Wellner JA (1996). Weak convergence and empirical processes. Springer.

Yeh I. and Lien C. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. Expert Systems with Applications, 36(2):2473–2480.

Zheng W, Balzer L, van der Laan MJ, Petersen M, and Collaboration S. (2018). Constrained binary classification using ensemble learning: an application to cost-efficient targeted PrEP strategies. Statistics in Medicine, 37(2):261–279. [PubMed: 28384841]

Zheng W. and van der Laan MJ (2011). Cross-validated targeted minimum-loss-based estimation. In Targeted Learning, pages 459–474. Springer.

Zou H. and Hastie T. (2005). Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B, 67(2):301–320.

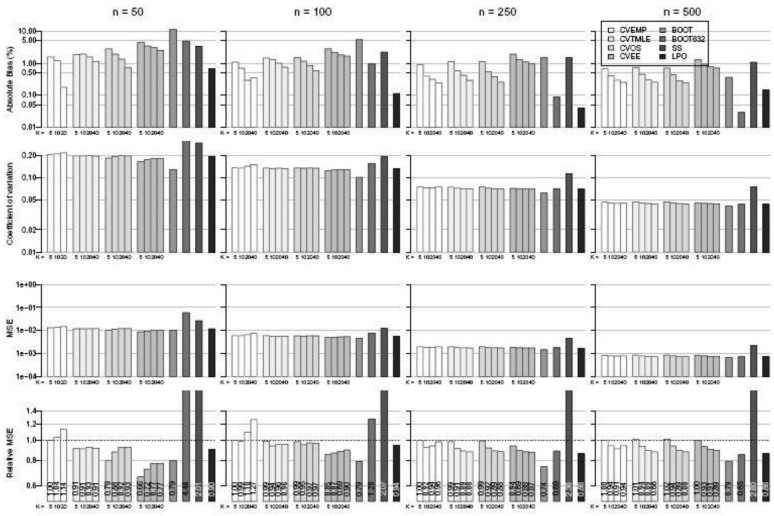

Fig. 1.

Performance of estimators of the AUC of a logistic regression. The vertical axis in each row is on a log-scale. Abbreviations: CVEMP = standard *K*-fold CV estimator; CVTMLE = *K*-fold CV targeted minimum loss-based estimator; CVOS = *K*-fold CV one-step estimator; CVEE = *K*-fold CV estimating equations estimator; BOOT = bootstrap corrected estimator; SS = sample-splitting estimator; LPO = leave-pair-out cross-validation estimator; MSE = mean squared-error.

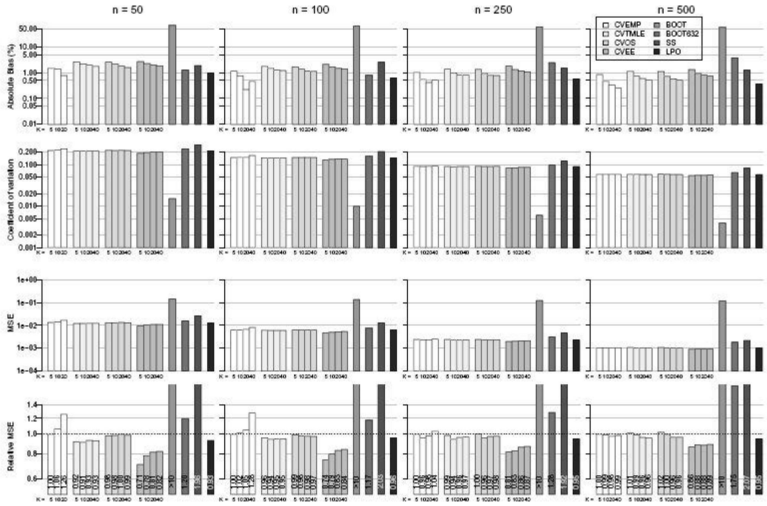

Fig. 2.

Performance of estimators of the AUC of a random forest. The vertical axis in each row is on a log-scale. Abbreviations: CVEMP = standard *K*-fold CV estimator; CVTMLE = *K*-fold CV targeted minimum loss-based estimator; CVOS = *K*-fold CV one-step estimator; CVEE = *K*-fold CV estimating equations estimator; BOOT = bootstrap corrected estimator; SS = sample-splitting estimator; LPO = leave-pair-out cross-validation estimator; MSE = mean squared-error.

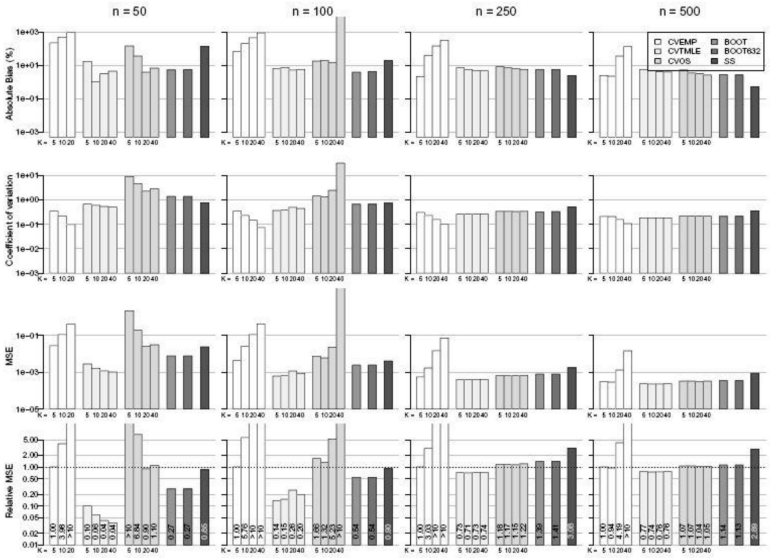

Fig. 3.

Performance of estimators of SCRNP of a logistic regression classifier with 95% sensitivity. The vertical axis in each row is on a log-scale. Abbreviations: CVEMP = standard *K*-fold CV estimator; CVTMLE = *K*-fold CV targeted minimum loss-based estimator; CVOS = *K*-fold CV one-step estimator; BOOT = bootstrap corrected estimator; MSE = mean squared-error.

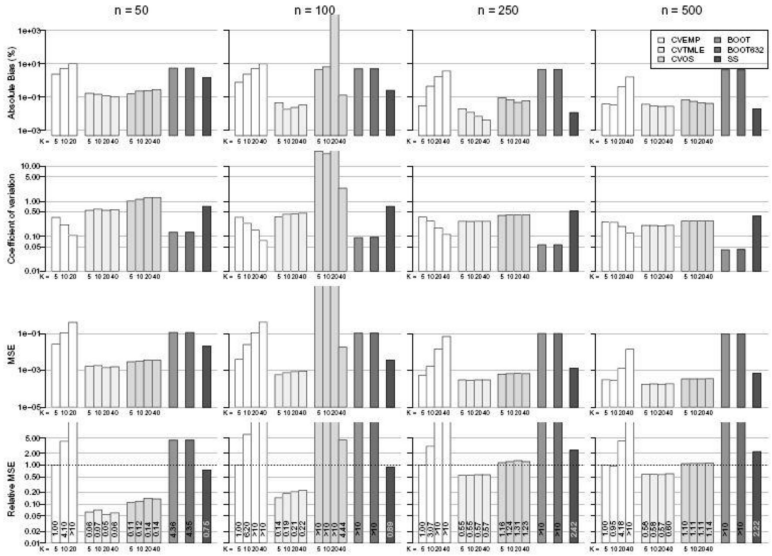

Fig. 4.

Performance of estimators of SCRNP of a random forest classifier with 95% sensitivity. The vertical axis in each row is on a log-scale. Abbreviations: CVEMP = standard *K*-fold CV estimator; CVTMLE = *K*-fold CV targeted minimum loss-based estimator; CVOS = *K*-fold CV one-step estimator; BOOT = bootstrap corrected estimator; MSE = mean squared-error.

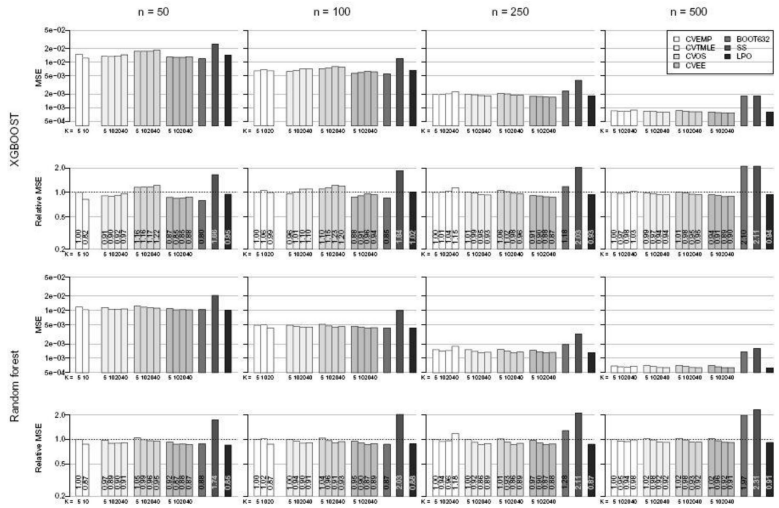

Fig. 5.

Performance of estimators of the AUC. The vertical axis in each row is on a log-scale. Abbreviations: CVEMP = standard *K*-fold CV estimator; CVTMLE = *K*-fold CV targeted minimum loss-based estimator; CVOS = *K*-fold CV one-step estimator; CVEE = *K*-fold CV estimating equations estimator; BOOT = bootstrap corrected estimator; LPO = leave-pair-out cross-validation estimator; MSE = mean squared-error.

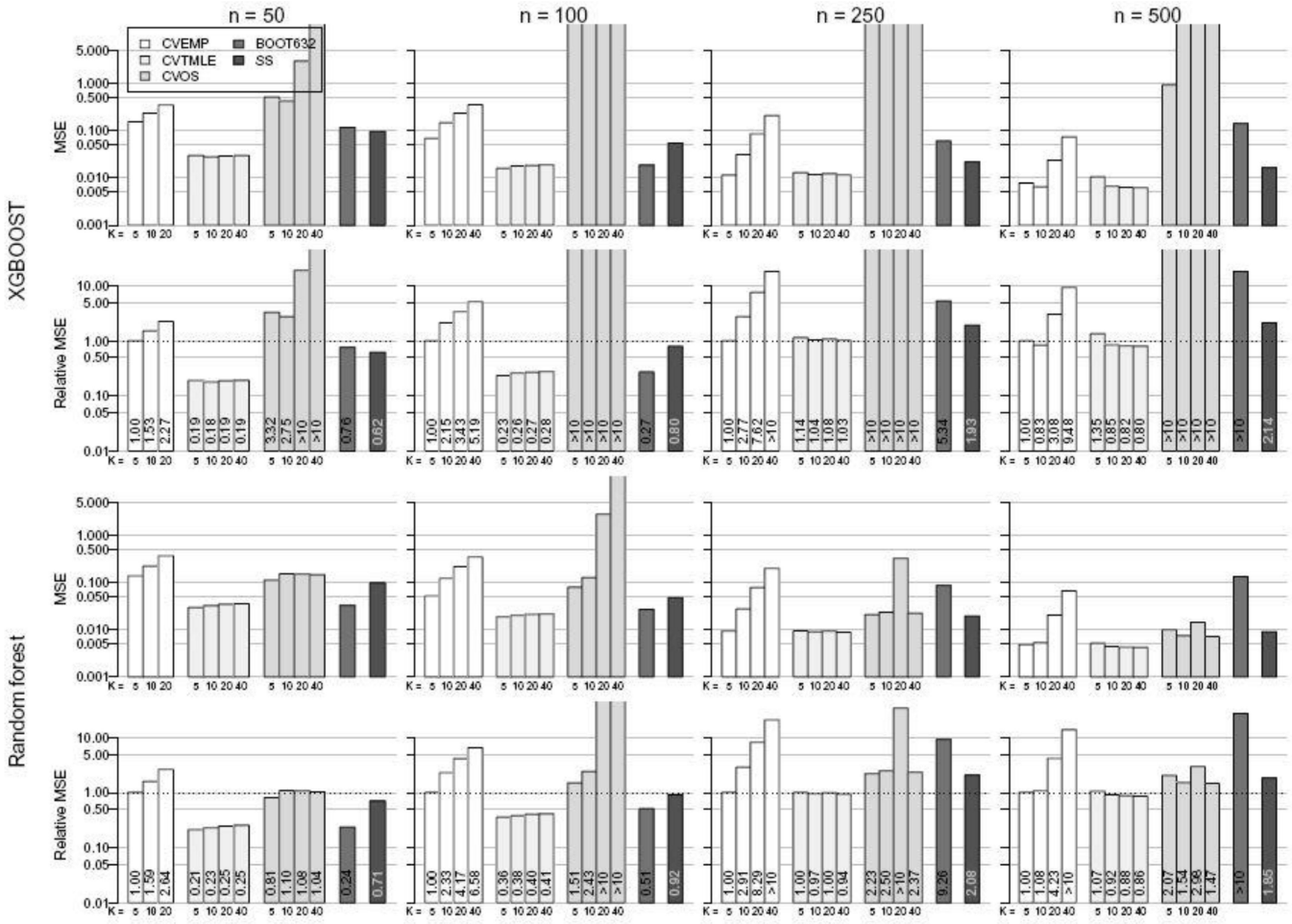

Fig. 6.

Performance of estimators of the SCRNP for a classifier with 95% sensitivity. The vertical axis in each row is on a log-scale. Abbreviations: CVEMP = standard *K*-fold CV estimator; CVTMLE = *K*-fold CV targeted minimum loss-based estimator; CVOS = *K*-fold CV one-step estimator

Table 1

Data sets analyzed. Notation: N = total number of observations in the data set; p = total number of covariates in the data set (note: categorical variables were categorized into multiple binary variables, each counting towards p); g_N = marginal probability $Y = 1$ in the entire data set.

Name	Citation	N	p	g_N
adult	Kohavi (1996)	32,561	86	0.24
bank	Moro et al. (2014)	41,188	54	0.11
cardio	Ayres-de Campos et al. (2000)	2,126	21	0.14
default	Yeh and Lien (2009)	30,000	23	0.22
drugs	Fehrman et al. (2017)	1,885	12	0.12
magic	Bock et al. (2004)	19,020	10	0.65
wine	Aeberhard et al. (1992)	6,497	12	0.20